

# Aula Um

## Histórico

- Primeiros computadores: grandes e caros
- Anos 50-60: *spooling*, multiprogramação
- Início dos anos 60: sistemas *time sharing*
- Final dos anos 60 início dos anos 70: surgimento de redes de computadores
- A partir dos anos 70 inicia-se a pesquisa em sistemas distribuídos

## Por que sistemas distribuídos ?

Com a melhoria das tecnologias o que se conseguia executar algumas décadas atrás somente com computadores que custavam milhões de dólares, hoje se consegue executar com computadores que custam poucos dólares. (e.g. 10 milhões de dólares para executar 1 instrução por segundo, para computadores que custam 1000 dólares e podem executar mais de 100 MIPS). Comparativo: se carros tivessem melhorado na mesma proporção no mesmo período um Rolls Royce custaria hoje 10 dólares e faria 1 bilhão de quilômetros com um litro de gasolina (Tanenbaum).

O segundo fator é o surgimento de redes de computadores de alta velocidade, onde informações podem ser transferidas entre computadores na faixa de microssegundos. (e.g. 1.28 Gbytes/s Myrinet).

Como resultado é possível conectar diversos computadores através de uma rede de alta velocidade para executar um sistema de computação. Estes sistemas são geralmente chamados *sistemas distribuídos* - *S.D.*, em contraste a *sistemas centralizados* - *S.C.* (ou sistemas de um único processador).

Existe um único problema em tudo isto: software. Programação em/de sistemas distribuídos é completamente diferente de sistemas centralizados, e sistemas operacionais distribuídos estão no seus primórdios.

## O que é um Sistema Distribuído

*Um sistema distribuído é uma coleção de computadores independentes que parecem como um único sistema para o usuário* (Tanenbaum).

Dois aspectos a serem considerados:

- hardware: as máquinas são autônomas.

- software: o usuário pensa no sistema como um computador único.

Exemplos:

- Uma fábrica cheia de robôs, cada um com um computador controlando suas funções (visão, comunicação ...). Quando um robô nota que uma peça tem um defeito, ele se comunica com outro robô para que aquela peça seja substituída. Se todos os robôs podem agir como se fossem periféricos conectados a um mesmo computador central e serem programados desta forma, então tal sistema conta como sendo um sistema distribuído.
- Um grande banco com diversas agências. Cada agência com um computador que controla as contas daquela agência e consegue se comunicar com os computadores de outras agências e com o computador central da agência matriz. Se as transações podem ser realizadas por um cliente independente da agência que ele esteja, e que ele não note a diferença entre este sistema e um sistema centralizado (como costuma(va) ser), então este também pode ser considerado um sistema distribuído.

## Objetivos

Só porque é possível de ser construído não significa necessariamente que um sistema distribuído é uma boa idéia. Exemplo, é possível colocar 4 *drives* para disquetes em um computador.

## Vantagens de S.D. sobre S.C.

- Economia: *Grosh's law: performance proporcional custo*<sup>2</sup>  
Funciona bem para *mainframes*. Não funciona para microcomputadores.
- Velocidade: existem situações que não se consegue construir um mainframe que atinja a performance que se alcança com um sistema composto por, por exemplo, 10000 CPUs atuais. Imagine que cada uma consiga executar 50MIPS. Performance total de 500000 MIPS. Para uma única CPU conseguir executar em tal velocidade deveria executar uma instrução a cada 0.002 nanoseg. Não existe computador que chegou perto desta performance e é muito pouco provável que algum dia exista. A teoria da relatividade de Einstein diz que nada pode se deslocar mais rápido que a velocidade da luz, que pode se deslocar somente 0.6mm em 2 nanoseg. Um computador que coubesse em um chip de 0.6mm de tamanho iria gerar tanto calor que derreteria.
- Algumas aplicações são naturalmente distribuídas. Exemplos: uma cadeia de supermercados; CSCW (*computer-supported cooperative work*); games.

- **Confiabilidade:** Se a carga de trabalho for distribuída entre diversos computadores, a falha em um deles não afetará todo o sistema. Idealmente, se 5% dos computadores estão "fora do ar" em um determinado momento o sistema deveria continuar trabalhando com 5% de perda de performance. Para aplicações críticas como reatores nucleares ou aviões, o uso de sistemas distribuídos pode ser uma consideração dominante.
- **Expansão:** se uma companhia prospera ela precisará aumentar seu poder de computação. Com mainframes as únicas soluções seriam trocar o mainframe por um maior (se existir) ou comprar mais um. Isto pode causar um problema para a companhia. Com sistemas distribuídos é possível adicionar mais um processador conforme as necessidades forem crescendo.

### **Vantagens de S.D. sobre PCs independentes**

- **Compartilhamento de dados:** permite que diversos usuários acessem a uma base comum de dados;
- **Compartilhamento de periféricos:** como impressoras coloridas que possuem um alto custo;
- **Comunicação:** fazer a comunicação entre pessoas mais fácil, e.g. usando correio eletrônico, que possui inúmeras vantagens sobre outros métodos tradicionais (telefone, papel), e.g. é mais rápido, não precisa que duas ou mais pessoas estejam disponíveis ao mesmo tempo, produz documentos que podem ser editados, ...
- **Flexibilidade:** melhor aproveitamento de computadores que não são tão utilizados.

### **Desvantagens de S.D.**

- **Software:** pouco software disponível
- **Rede:** pode saturar ou causar problemas
- **Segurança:** Fácil acesso também se aplica a dados secretos

Mesmo assim, S.D. estão se tornando cada vez mais frequentes.

## **Modelos para Construção de Sistemas Distribuídos**

Segundo [Pradeep 1997]:

- Rede de minicomputadores
- Rede de estações de trabalho

- Rede de estações de trabalho com estações servidoras
- *Pool* de processadores

Segundo [Tanenbaum 1995] (SISD, SIMD, MISD (\*), **MIMD**):

- Computadores Paralelos e Distribuídos
  - Multiprocessadores (memória compartilhada)
    - \* Barramento
    - \* Chaveamento
  - Multicomputadores (memória privada)
    - \* Barramento (estações de trabalho em uma rede local)
    - \* Chaveamento (*Transputer*)